



Colour Swatches

Introduction and Overview

A colour swatch is a gadget used to display an example colour and allow the user to select another :

The Colour Swatch gadget
The Colour
Swatch gadget

The colour swatch can display any 24bit RGB colour or a 'None' value. The gadget requires the Wimp 3.98 or later to function correctly.

A colour swatch has a pop-up colour menu placed 8 OS Units to the right of the display area. The display area will show the currently selected colour as its filled background. On selecting the pop-up menu icon, a ColourDBox will be displayed, allowing the selection of another colour. Once a selection has been made and confirmed, the display will be updated and a notification message sent to the application.

The application can decide whether a 'None' option is available on the colour selector. If the 'None' option is available and selected, the display icon will show the text 'None' and the user will be notified of this. Where possible, the interface has been kept similar to that of the ColourDBox.

Bits in the flags word for a colour swatch gadget have the following meanings:

Bit(s) Meaning

- 0 Reserved, must be zero
- 2 when set, include a **None** button in the dialogue box
- 3 when set, select the **None** button when the dialogue box is created
- 4-31 Reserved, must be zero

Colour Swatch templates

Field	Size in bytes	Type
colour	4	word
title	4	MsgReference

Toolbox methods

ColourSwatch_GetColour (Method &1403000)

Read the colour of the Colour Swatch gadget

On entry

R0 = flags
R1 = Window object id
R2 = &1403000
R3 = Gadget component id

On exit

R0 = Whether the 'None' option is selected :

Value	Meaning
-------	---------

0	'None' is not selected
---	------------------------

1	'None' is selected
---	--------------------

R1 = Colour number in the form &BBGRR00

R2 - R9 preserved

Use

This method is used to read the colour currently in use by the ColourSwatch gadget.

Declarations

```
extern _kernel_oserror *colourswatch_get_colour ( unsigned
int flags,
ObjectId window,
ComponentId swatch,
int *outflags,
unsigned long *colour
);
```

ColourSwatch_SetColour (Method &1403001)

Set the colour of the Colour Swatch gadget

On entry

R0 = flags :

Bit(s) Meaning

0 Select the 'None' option

R1 = Window object id

R2 = &1403001

R3 = Gadget component id

R4 = Colour number in the form &BBGRR00

On exit

R0 - R9 preserved

Use

This method is used to change the colour currently in use by the ColourSwatch gadget.

Declarations

```
extern _kernel_oserror *colourswatch_set_colour ( unsigned
int flags,
ObjectId window,
ComponentId swatch,
unsigned long colour
);
```

ColourSwatch_GetTitle (Method &1403002)

Read the title of the colour selector used by the Colour Swatch gadget

On entry

R0 = flags
R1 = Window object id
R2 = &1403002
R3 = Gadget component id
R4 = pointer to buffer
R5 = size of buffer

On exit

R5 = size of buffer required (if R4 was 0)
else buffer pointed at by R4 contains text
R5 holds number of bytes written to buffer

Use

This method is used to read the title used on the ColourPicker dialogue which is displayed when the pop-up menu icon is selected.

Declarations

```
extern _kernel_oserror *colourswatch_get_title ( unsigned
int flags,
ObjectId window,
ComponentId swatch,
char *buffer,
int buff_size,
int *nbytes
);
```

ColourSwatch_SetTitle (Method &1403003)

Change the title of the colour selector used by the Colour Swatch gadget

On entry

R0 = flags
R1 = Window object id
R2 = &1403003
R3 = Gadget component id
R4 = pointer to string to use

On exit

None

Use

This method is used to change the title used on the ColourPicker dialogue which is displayed when the pop-up menu icon is selected. Any active ColourPicker for this gadget will be closed when this method is used.

Declarations

```
extern _kernel_osererror *colourswatch_set_title ( unsigned
int flags,
ObjectId window,
ComponentId swatch,
char *title
);
```

ColourSwatch_GetNoneAvailable (Method &1403004)

Read whether the 'None' option is available on the colour selector used by the ColourSwatch

On entry

R0 = flags
R1 = Window object id
R2 = &1403004
R3 = Gadget component id

On exit

R0 = flags:

Bit(s)	Meaning
0	None is available

Use

This method returns whether the **None** option appears in a colour selector window used when the pop-up menu is selected.

Declarations

```
extern _kernel_oserror *colourswatch_get_none_available (  
    unsigned int flags,  
    ObjectId window,  
    ComponentId swatch,  
    int *out_flags  
);
```

ColourSwatch_SetNoneAvailable (Method &1403005)

Set whether the 'None' option is available on the colour window

On entry

R0 = flags
R1 = Window object id
R2 = &1403005
R3 = Gadget component id
R4 = non-zero means **None** is available

On exit

R1 - RR9 preserved

Use

This method sets whether a **None** option appears in the colour selector displayed when .

Declarations

```
extern _kernel_oseerror *colourswatch_set_none_available (  
    unsigned int flags,  
    Objectid window,  
    Componentid swatch,  
    int none  
);
```

Toolbox events

ColourSwatch_ColourChanged (Event &140300)

Notification that the colour of the ColourSwatch has been changed

Message

Offset Contents

R1+8 @140300

R1+12 flags:

Bit(s) Meaning

0 the 'None' option was selected

R1+16 colour in the form @BBGRR00

Use

This Toolbox event is raised when the colour of the ColourSwatch has changed. If the 'None' option was available and selected, then bit 0 of the flags word is set.

Declarations

```
typedef struct
{
    ToolboxEventHeader hdr;
    int new_colour;
} ColourSwatchColourChangedEvent;
```

Document information

Maintainer(s): RISCOS Ltd <developer@riscos.com>

History:

Revision	Date	Author	Changes
----------	------	--------	---------

1	02 May 2004	ROL	Initial version
---	-------------	-----	-----------------

2	06 May 2004	ROL	Corrected reference
---	-------------	-----	---------------------

- The gadget format uses a MsgReference, not a StringReference, 'cos it's user visible.

Disclaimer: Copyright © Pace Micro Technology plc, 2001.

Portions copyright © RISCOS Ltd, 2001-2004.

Published by RISCOS Limited.

No part of this publication may be reproduced or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, or stored in any retrieval system of any nature, without the written permission of the copyright holder and the publisher, application for which shall be made to the publisher.
